

## PATENT ABSTRACTS OF JAPAN

(11)Publication number : 05-035872

(43)Date of publication of application : 12.02.1993

(51)Int.Cl.

G06F 15/70

G06F 15/66

G06F 15/66

(21)Application number : 03-214716

(71)Applicant : VICTOR CO OF JAPAN LTD

(22)Date of filing : 31.07.1991

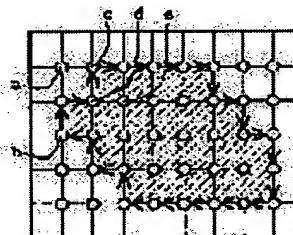
(72)Inventor : HOZUMI YOSHIKO

## (54) CONTOUR TRACING SYSTEM FOR BINARY IMAGE

## (57)Abstract:

PURPOSE: To perform high-speed processing and accurately reproduce an image.

CONSTITUTION: The tracing direction of a border point is defined according to whether four picture elements at the periphery of the border point of a picture element are black or white. Namely, the tracing direction is set on a border line where a black picture element is present on the right side and a white picture element is present on the left side; and '1' is set for an upward tracing direction, '2' for a leftward tracing direction, '3' for a downward tracing direction, and '0' for a rightward tracing direction. Further, the respective tracing directions are assigned to the respective bits of a 4-bit tracing direction flag and the bits are set to '1' when there is the tracing direction or to '0' when not. Then all border points of the image are given tracing direction flags and the border points are retrieved according to the tracing direction flag and regarded as origins and while tracing is performed from the origins in the tracing directions, the tracing direction flags at the border points which are already traced are changed; and thus the tracing is carried on all the tracking direction flags show '0000' to find the contour of the image.



## LEGAL STATUS

[Date of request for examination] 21.09.1994

[Date of sending the examiner's decision of rejection] 28.10.1998

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

(19)日本国特許庁(J P)

(12) 公 開 特 許 公 報 (A)

(11)特許出願公開番号

特開平5-35872

(43)公開日 平成5年(1993)2月12日

(51)Int.Cl. <sup>5</sup>	識別記号	庁内整理番号	F I	技術表示箇所
G 0 6 F 15/70	3 4 0	9071-5L		
15/66	3 3 0 Q	8420-5L		
	4 1 0	8420-5L		

審査請求 未請求 請求項の数1(全 6 頁)

(21)出願番号 特願平3-214716

(22)出願日 平成3年(1991)7月31日

(71)出願人 000004329

日本ビクター株式会社

神奈川県横浜市神奈川区守屋町3丁目12番  
地

(72)発明者 穂積 芳子

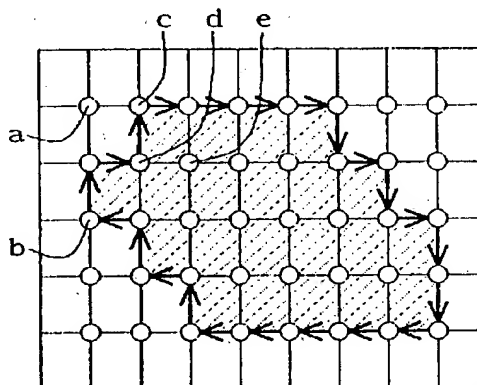
神奈川県横浜市神奈川区守屋町3丁目12番  
地 日本ビクター株式会社内

(54)【発明の名称】 2値画像の輪郭追跡方式

(57)【要約】

【目的】 高速に処理をおこない、かつ、正確な画像の再生を可能にする。

【構成】 画素の境界点の周囲4画素の白／黒によってその境界点の追跡方向を定義する。即ち、右側に黒画素、左側に白画素が来る境界線上に追跡方向を定め、それぞれこの追跡方向が上方向であるときを「1」、左方向であるときを「2」、下方向であるときを「3」、右方向であるときを「0」とし、それぞれの追跡方向を4ビットの追跡方向フラグの各ビットに割り当てて、追跡方向があるビットを「1」、ないビットを「0」と定義する。そして、画像の全ての境界点にこの追跡方向フラグを付け、追跡方向フラグから境界点を検索して起点とし、この起点から追跡方向に添って追跡すると共に、追跡済みの境界点の追跡方向フラグを変更して行き、全ての追跡方向フラグが「0000」になるまで追跡を行うと、画像の輪郭を求めることができる。



## 【特許請求の範囲】

【請求項1】 2値画像の輪郭となる境界画素のいくつかを抽出してベクトル化を行う2値画像の輪郭追跡方式において、画素の境界点における4方向追跡コードを前記2値画像の画素の境界点に付し、この追跡コードを追跡することにより得られる境界画素の境界点を用いてベクトル化を行うことを特徴とする2値画像の輪郭追跡方式。

## 【発明の詳細な説明】

## 【0001】

【産業上の利用分野】 本発明は、画素の集合（ドットパターン）として構成された画像（文字パターン）の輪郭を抽出してベクトル化を自動的に行う2値画像の輪郭追跡方式に関するものである。

## 【0002】

【従来の技術】 従来より、スキャナ等により読取った文字をアウトラインデータに変換して使用したり、ROM等の記憶装置に格納されたドットパターンとして構成された文字パターンを拡大・縮小して使用するために、文字パターンの輪郭をベクトル化処理している。このベクトル化を自動的に行う2値画像の輪郭追跡方式は、従来から種々のものが考えられている。例えば、特開昭62-131382号及び特開昭64-91176号には、入力された2値画像を縦横各2倍に拡大してから、画素が黒（1）か白（0）かを判別して、輪郭を追跡するための起点を検索し、さらに、この起点となる画素に接する上下左右斜めの8画素を時計回りに調べて、画素が0から1に変化した方向に移動することにより、輪郭となる境界画素の中心を追跡する方式が開示されている。また、特開昭64-6889号には、境界画素の中心を追跡するチェーンコード符号列から図形の輪郭の折れ曲り状態を判定して直角部を検出し、近傍の画素をベクトルポイントとして選択する方式が開示されている。

## 【0003】

【発明が解決しようとする課題】 従来の2値画像の輪郭追跡方式において、2値画像を縦横各2倍に拡大してから、輪郭を追跡している場合には、4倍の画像データ量を処理しなければならず、大容量のメモリと処理時間とを必要としていた。そして、境界画素の中心を追跡しているのに、元の画像と再生後の画像とでアウトライン（輪郭）に誤差が生じることがあった。また、起点を検索したり、境界画素の中心を追跡する際に、毎回画素が0から1に変化する方向を調べているので、処理量が多くなり、時間がかかるという課題があった。そして、境界画素の中心を追跡するチェーンコード符号列を用いた場合には、一画素幅の直線が突出しているときには、一つの境界画素の中心を二度対応させることになるので、文字を拡大した際にもこの直線は一画素幅のままとってしまうという問題点があった。そこで本発明は、データ量を少なくして処理時間を短くし、さらにアウトライ

ンに生じる誤差を少なくした2値画像の輪郭追跡方式を提供することを目的とする。

## 【0004】

【課題を解決するための手段】 上記目的を達成するための手段として、2値画像の輪郭となる境界画素のいくつかを抽出してベクトル化を行う2値画像の輪郭追跡方式において、画素の境界点における4方向追跡コードを前記2値画像の画素の境界点に付し、この追跡コードを追跡することにより得られる境界画素の境界点を用いてベクトル化を行うことを特徴とする2値画像の輪郭追跡方式を提供しようとするものである。

## 【0005】

【実施例】 本発明の2値画像の輪郭追跡方式の一実施例を図面と共に説明する。まず、ある画素の境界点の周囲4画素の白／黒各画素の境界線上にその境界点の追跡方向を定義する。即ち、右側に黒画素、左側に白画素が来る境界線上に、図1（A）に示すような追跡方向（追跡コード）を定め、それぞれこの追跡方向が上方向であるときを「1」、左方向であるときを「2」、下方向であるときを「3」、右方向であるときを「0」とする。そして、それぞれの追跡方向を図1（B）に示すように4ビットの追跡方向フラグの各ビットに割り当てて、追跡方向があるビットを「1」、ないビットを「0」と定義する。このようにして定義すると画素の境界点（丸印）に接する黒画素（斜線部）と白画素との組み合わせは、図2（A）～（P）に示すように全部で16通りとなり、1つの境界点における追跡方向の数は、0、1方向または2方向のいずれかとなる。このとき、この境界点が境界画素の境界点であるならば、必ず、周囲4画素に白と黒の両方が混在するので、図2（A）の4方向全てが白画素の場合と図2（P）の4方向全てが黒画素の場合だけが、境界画素の境界点でないことになる。そして、この図2（A）、（P）の場合は、追跡方向がない場合であり、追跡方向フラグは「0000」となっている。したがって、この追跡方向フラグ「0000」以外の追跡方向フラグを有する境界点が境界画素の境界点となるので、これらの追跡方向フラグを検索すれば良いことになる。

【0006】 ここで、図3に示すような画像をベクトル化する場合について説明する。なお、斜線部は黒画素であり、境界点は丸印で示した。まず、同図において、全ての境界点に図2に示した追跡方向フラグを付ける。例えば、図中、境界点aは、周囲4画素全てが白画素であるので、図2（A）の追跡フラグ「0000」が付けられ、境界点bは、右上だけが黒画素であるので、図2（B）の追跡フラグ「0100」が付けられる。同様に、境界点cは図2（I）の「1000」、境界点dは図2（N）の「0100」、境界点eは図2（P）の「0000」がそれぞれ付けられる。次に、追跡方向フラグが「0000」でない境界点を検索し、そのうち一

3

つを起点とする。そして、図4に示すように、この起点から追跡方向に添って追跡すると共に、追跡済みの境界点の追跡方向フラグの追跡方向に相当するビット(図1(B)参照)をリセットし、「0」に変更して行く。即ち、境界点dでは、追跡方向は上方向の「1」であるので、境界点d上にある境界点cに移動する。そして、同時に、境界点dの追跡フラグ「0100」の上方向に相当するビット(左から2番目のビット)を「0」に変更し、「0000」とする。同様に、境界点cの追跡方向は右方向の「0」であるので、境界点cの右隣の境界点

に移動すると共に、追跡フラグ「1000」の右方向に相当するビット(左から1番目のビット)を「0」に変更し、「0000」とする。

【0007】このようにして、追跡済みの境界点の追跡方向フラグの追跡方向に相当するビットを「0」に変更して行き、全ての追跡方向フラグが「0000」になるまで追跡を行うことにより、画像の輪郭を求めることができる。そして、この境界点の追跡方向を元にしてベクトル化を行うことができる。

【0008】次に、図5(A)に示すように、境界点fにおいて画素が斜めに接している場合の追跡方向について説明する。図5(A)に示す画像を上述したのと同様にして全ての境界点の追跡方向を検索すると、図5

(B)のようになり、境界点fは、図2(K)に示す2方向の追跡方向を有する追跡方向フラグ「1010」が付される。このとき、追跡方向は、2方向であるので、追跡されてきた方向によってどちらか一方に追跡するようにしなければならない。そして、この追跡する方向は、次に示す2通りの方向が考えられる。

【0009】まず、8方向に連結される画素を一つの画像とみなす(斜め接しを一つの画像として認める)場合は、図6(A)、(B)に示すように、2方向の追跡方向を有する境界点において、常に反時計回り(左回り)となるように追跡すると、斜に接した画素の周囲を回って追跡するので、一つの画像として認識される。即ち、図5(B)において、境界点fは図6(B)の追跡方向が適用されるので、境界点gから境界点fに追跡されてきたときは、境界点iの方向に追跡されて右下の画像を一周し、境界点hから境界点fに追跡されてきたときは、境界点jに追跡されるので、図5(B)の画像は1つの画像として認識されることになる。そして、境界点fの追跡方向フラグは「1010」であるので、境界点gから境界点fを通過して境界点iの方向に追跡されるときには、境界点fの追跡フラグ「1010」の右方向に相当するビット(左から1番目のビット)を「0」に変更し、この時点で、「0010」となる。その後、境界点hから境界点fを通過して境界点jの方向に追跡されるときには、境界点fの追跡フラグ「0010」の左方向に相当するビット(左から3番目のビット)を「0」に変更し、「0000」となる。

4

【0010】また、4方向に連結される画素のみを一つの画像とみなす(斜め接しを一つの画像として認めない)場合は、図7(A)、(B)に示すように、2方向の追跡方向を有する境界点において、常に時計回り(右回り)となるように追跡すると、斜に接した画素を無視して追跡することになるので、斜めに接した画素同士は、別々の画像として認識される。これを図5(B)において考えると、図7(B)が適用されるので、境界点gから境界点fに追跡されてきたときは、境界点jの方向に追跡されることになり、右下の画像へ連続して追跡しないので、左上の画素だけで1つの画像と認識される。また、境界点hから境界点fに追跡されてきたときは、境界点iに追跡されて右下の画像もこれで1つの画像として認識されるので、結局、図5(B)の画像は2つの画像として認識されることになる。そして、この境界点gから境界点fを通過して境界点jの方向に追跡されるときには、境界点fの追跡フラグ「1010」の左方向に相当するビット(左から3番目のビット)を「0」に変更し、この時点で、「1000」となる。その後、境界点hから境界点fを通過して境界点iの方向に追跡されるときには、境界点fの追跡フラグ「1000」の右方向に相当するビット(左から1番目のビット)を「0」に変更し、「0000」となる。

【0011】具体例として、図8に示す画像を追跡した結果を図9～図12に示す。図9は、図8の画像から上記実施例を使用して境界画素の境界点のみを抽出し、この全境界点を追跡したものであり、これを許容誤差を1.5ドット(画素)としてベクトル化したものが図10である。また、同様に、図11は、図8の画像を従来例で説明した画素の中心を追跡する方式によって境界画素を抽出し、この全境界画素の中心を追跡したものであり、これを許容誤差を1.5ドットとしてベクトル化したものが図12である。本実施例と従来例とを比較してみると、本実施例は黒画素と白画素の境界となる境界点を追跡しているので、全境界点を追跡したときに(図9)、図8の元の画像の輪郭とほぼ同じ輪郭が得られるが、従来例では、黒画素の中心を追跡しているので(図11)、図8に示す元の画像よりもやや太くなっている。このことは、図9、の部分Aと図11の部分Bのところでその差が顕著に現れている。

【0012】以上説明したように、本発明は、入力された画像に予め定義した追跡方向を当てはめていくだけで境界画素(境界点)を追跡できるので、毎回追跡方向を検索する必要はなく、処理時間が速くなる。また、画素を縦横各2倍に拡大する(4倍のデータ量を扱う)ことなく追跡することができるので、大容量のメモリは不要であり、処理時間も短くて済む。なお、上記実施例では、右回りで追跡する例を示したが、図1、2で定義した追跡方向を、右側に白画素、左側に黒画素が来るように定義すれば左回りで追跡することができる。

## 【0013】

【発明の効果】本発明の2値画像の輪郭追跡方式は、境界画素の境界点を追跡するようにしたので、ベクトル化を行うための境界点の座標は、元の画像の実際の輪郭と同一となり、正確な画像の再生が可能となる。また、定義する追跡方向は少なく、単純であり、特殊な場合を考慮する必要がないので、高速に処理することができる。そして、境界画素の境界点を追跡しているのに、一面素幅の直線が突出しているときでも追跡コードが同じところを戻ることがなく、文字を拡大したときに、この直線が一面素幅のままとならずに線幅も拡大することができるという効果がある。

## 【図面の簡単な説明】

【図1】(A)は本発明の2値画像の輪郭追跡方式に使用する追跡コードの一実施例を示す概念図であり、(B)はその追跡方向フラグである。

【図2】(A)～(P)は図1に示した追跡コードと追跡方向フラグとの対応関係を示す概念図である。

【図3】2値画像の一例を示す概念図である。

【図4】図2に示した2値画像を本発明の輪郭追跡方式に使用する追跡コードによって追跡した一実施例を示す概念図である。

【図5】(A)は2値画像の他の例を示す概念図であ

り、(B)はその2値画像を本発明の追跡コードによって追跡した一実施例を示す概念図であり、(C)はその一部拡大図である。

【図6】(A)、(B)は追跡方向が2方向ある場合に8方向に連結される画素を同一画像とするときの追跡方向を示す概念図である。

【図7】(A)、(B)は追跡方向が2方向ある場合に4方向に連結される画素を同一画像とするときの追跡方向を示す概念図である。

【図8】2値画像の一例を示す概念図である。

【図9】図8に示した2値画像を本発明の輪郭追跡方式に使用する追跡コードによって抽出した全境界点を追跡したものを示す概念図である。

【図10】図9に示したものをベクトル化した一実施例を示す概念図である。

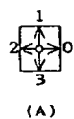
【図11】図8に示した2値画像を従来の方式によって抽出した全境界画素の中心を追跡したものを示す概念図である。

【図12】図11に示したものをベクトル化した一実施例を示す概念図である。

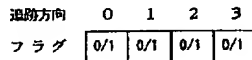
## 【符号の説明】

a～j 境界点

【図1】

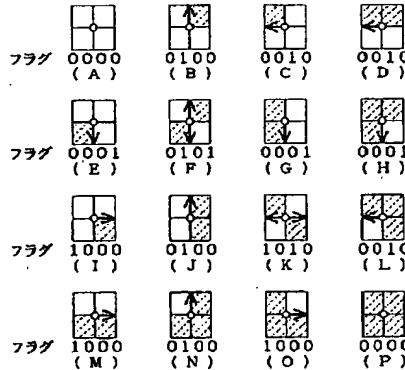


(A)

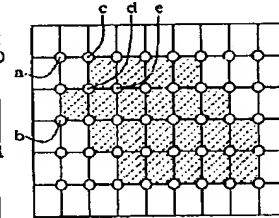


(B)

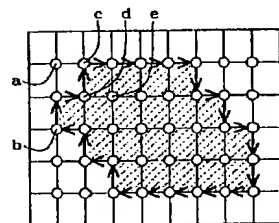
【図2】



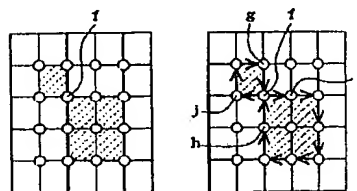
【図3】



【図4】



【図5】

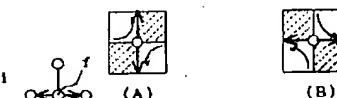


(A)

(B)

(C)

【図6】



(B)

【図7】



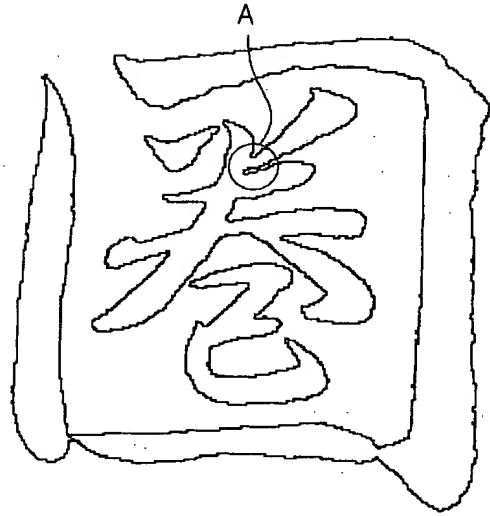
(A)

(B)

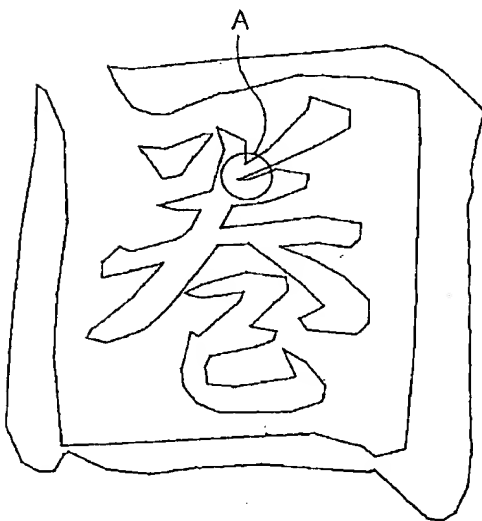
【図8】



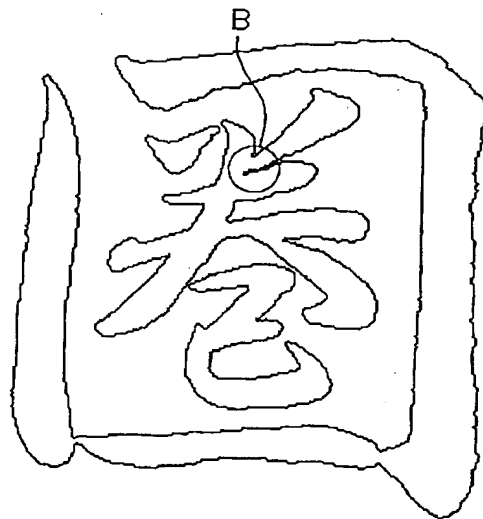
【図9】



【図10】



【図11】



( 6 )

特開平5-35872

【図12】

